ECE 517: Machine Learning

Assignment 6.2: Dual Formulation of a Nonlinear Classifier With MMSE

David Kirby – 101652098 – davidkirby@unm.edu

Fall 2021

SCHOOL OF ENGINEERING

DUAL FORMULATION OF A NONLINEAR CLASSIFIER WITH MMSE

Use the functions of the previous assignment to reconstruct the example of lesson 6.1 but using a dual representation and the polynomial kernel $k(\mathbf{x}_i, \mathbf{x}_j) = (\mathbf{x}_i^\top \mathbf{x}_j + 1)^3$.

1. Construct a train dataset and represent them.
2. Construct a function that computes the kernel matrix $\mathbf{K}$.
3. Compute the dual weights α_i of the MMSE solution.
4. Write an estimator in dual form as a function of kernel dot products between the training and test data.
5. Plot the boundary.
6. Repeat the experiment, but using the Ridge Regression solution, this is

$$\alpha = (\mathbf{K} + \gamma\, \mathbf{I})^{-1}\, y$$

   where γ is a small number. Show the result for different values of the parameter that are able to produce different solutions. Comment the results.

Provide a document that summarizes the theory and a graph of the result. Comment your results.

Dual formulation of a non-linear classifier and the polynomial kernel.

To solve the curse of dimensionality, we use the previous equation from Assignment 6.1:

$$\mathbf{w} = (\Phi\Phi^\top)^{-1}\Phi y \tag{1}$$

In primal notation, $\Phi\Phi^\top$ is the autocorrelation matrix $\mathbf{R}$.

Since here we only need a bunch of dot products, we find a dot product that we can easily compute without previously computing the transformation into the Hilbert space. By doing this dot product, we are passing Φ that has a number of dimensions into a dual space that has N dimensions because here we have N samples. This dual representation keeps the same properties as the primal space.

We will call $\Phi^\top\Phi$ the kernel $\mathbf{K}$, and it is a matrix that contains all the dot products between the data. This kernel trick will allow us to pass from a problem that has a matrix with dimensions equal to the number of dimensions of the space (e.g., 10 or 56) and use a matrix that has dimensions $N \times N$, where N is the number of data.

The next step consists of finding a dot product in this higher-dimensional space that can be expressed as a function of the input space only. For our example with the order three Volterra, this dot product is:

$$< \phi(\mathbf{x}_n), \phi(\mathbf{x}_m) >= (\mathbf{x}_n^\top \mathbf{x}_m + 1)^3 \tag{2}$$

We can also use Mercer's Theorem, which states that if $k(\mathbf{x}_n, \mathbf{x}_m)$ exists, gives a real number, and is positive definite, then it is a dot product in the Hilbert space and we can say:

$$k(\mathbf{x}_n, \mathbf{x}_m) =< \phi(\mathbf{x}_n), \phi(\mathbf{x}_m) > \tag{3}$$

This dot product in the Hilbert space can be expressed as a function of the vectors in the input space, no longer dependent on Φ. We can now plug this function into the previous estimator. Thus, using the Representer Theorem and finding a kernel dot product in a space of higher dimension, we solve the curse of dimensionality and then a nonlinear estimator can be constructed by a nonlinear transformation. This allows us to have spaces with a high dimensionality, even infinite, and we don't even notice because we don't go inside the Φ dot product.

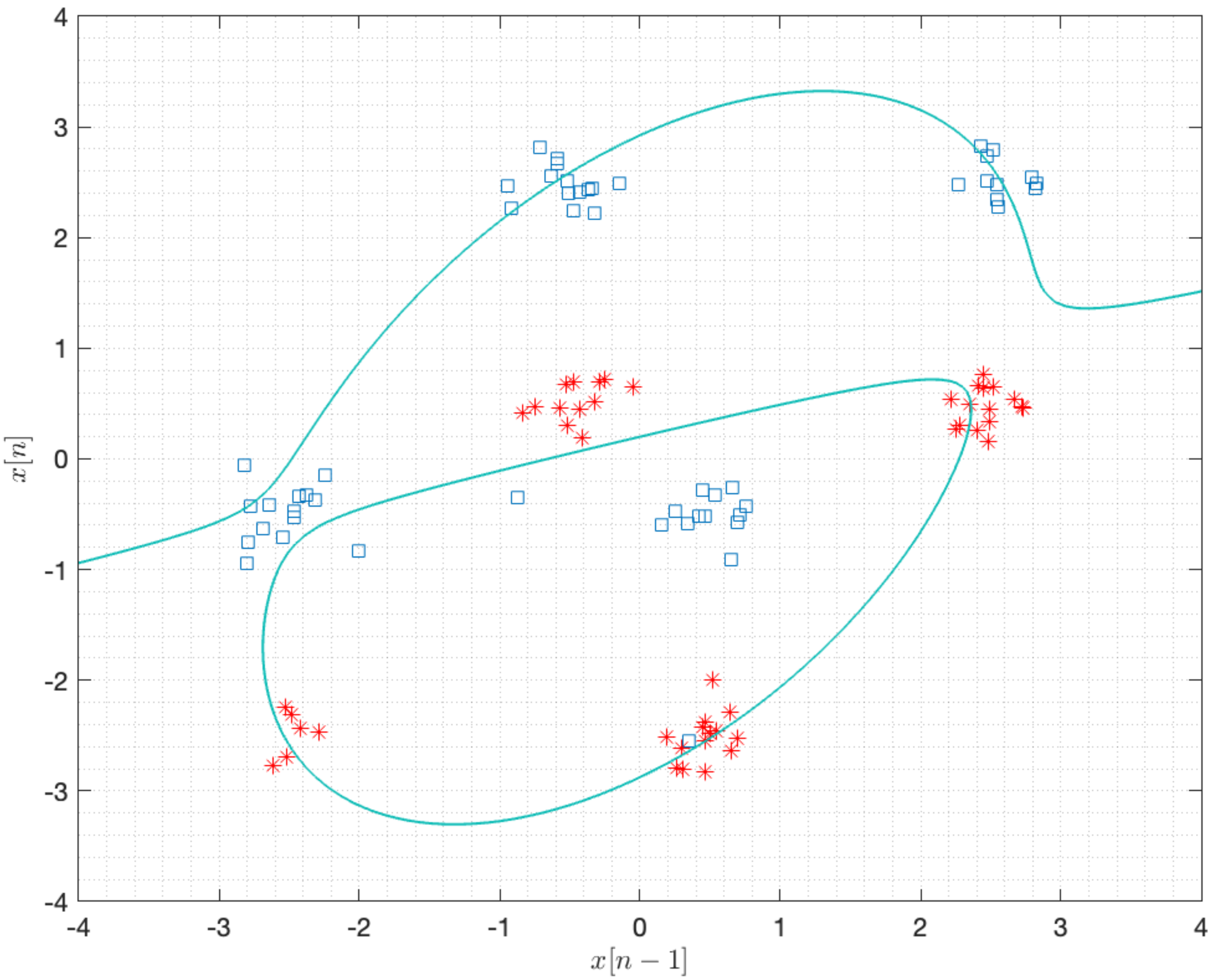

Figure 1: Dual representation with the polynomial kernel $k(\mathbf{x}_i, \mathbf{x}_j) = (\mathbf{x}_i^\top \mathbf{x}_j + 1)^3$.

Ridge Regression solution.

Using the properties of kernels, we can expand to Ridge Regression. By comparing figures 1 and 2 we can see that the ridge regression algorithm appears to give a more accurate fit.

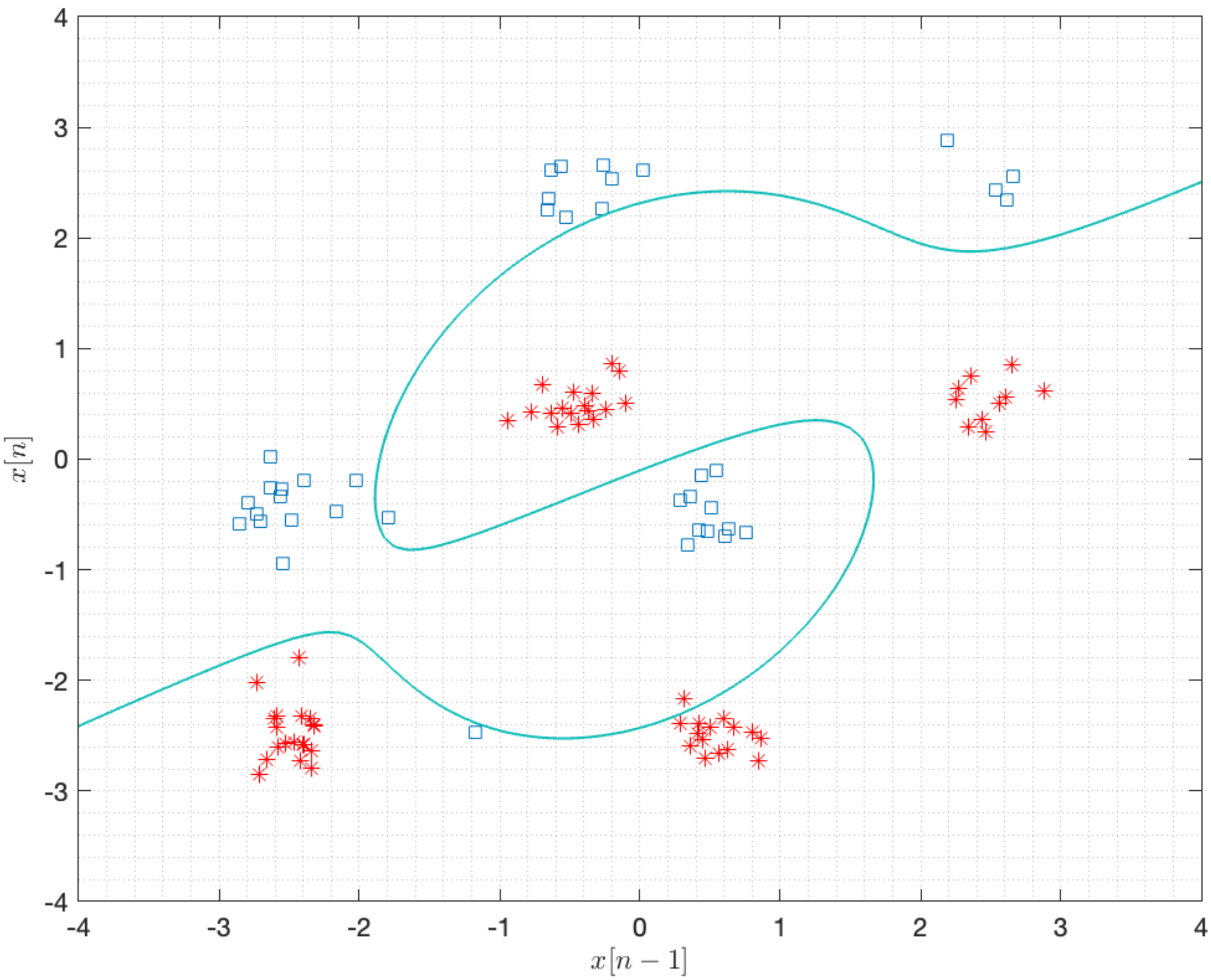

Figure 2: Ridge regression solution using $\alpha = (\mathbf{K} + \gamma\, \mathbf{I})^{-1}\, y$.